

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Luka Zlatečan

Razvoj mobilne aplikacije Vitis

DIPLOMSKO DELO

INTERDISCIPLINARNI UNIVERZITETNI
ŠTUDIJSKI PROGRAM PRVE STOPNJE
RAČUNALNIŠTVO IN MATEMATIKA

MENTOR: prof. dr. Marko Bajec

Ljubljana, 2017

COPYRIGHT. Rezultati diplomske naloge so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavo in koriščenje rezultatov diplomske naloge je potrebno pisno privoljenje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

V diplomskem delu razvijte mobilno aplikacijo, ki bo omogočala identifikacijo nevarnosti za razvoj različnih bolezni na vinski trti ter ocenjuje prisotnosti škodljivcev trte. Vir podatkov naj bodo vremenske postaje, ki omogočajo zajem ustreznih podatkov za napovedovanje tovrstnih nevarnosti.

Rad bi se zahvalil svoji družini, predvsem mami in starim staršem, prijateljem, ki so me podpirali na poti do zaključka študija, in Alešu Germovšku, ki je idejni vodja projekta.

Moji Veroniki.

Kazalo

Povzetek

Abstract

1	Uvod	1
1.1	Glivične bolezni vinske trte	1
1.2	Škodljivci vinske trte	2
1.3	Nadzor	3
2	Pridobivanje podatkov	5
2.1	WeatherLink	5
2.2	IPM modul za grozdje	9
3	Zasnova in tehnični opis rešitve	11
3.1	Skripta za izvoz	12
3.2	Skripta za razčlenjenje	13
3.3	Podatkovna baza	13
3.4	API	19
3.5	Mobilna aplikacija	20
3.6	Uporabljena orodja in tehnologije pri razvoju	20
4	Opis delovanja rešitve	25
4.1	API	26
4.2	Mobilna aplikacija	35

5 Sklep	47
Literatura	50

Seznam uporabljenih kratic

kratica	angleško	slovensko
SDK	software development kit	programska oprema za razvoj programske opreme
HTML	hypertext markup language	hipertekstni označevalni jezik
CSS	cascading style sheets	kaskadni listi slogov
CLI	comand line interface	vmesnik ukaznega poziva
SQL	structured query language	strukturirani povpraševalni jezik
API	application programing interface	aplikacijski programski vmesnik
HTTP	hyper text transfer protocol	protokol prenosa hiperpovezave
VCS	version control system	sistem za nadzor različic
IPM	integrated pest management	integrirano zatiranje škodljivcev
ID	identification	identifikacija
SCP	secure copy	varno kopiranje
REST	representational state transfer	predstavitveni prenos stanja

Povzetek

Naslov: Razvoj mobilne aplikacije Vitis

Avtor: Luka Zlatečan

Diplomska naloga razlaga razvoj mobilne aplikacije Vitis. Aplikacija bo prikazovala podatke o boleznih in škodljivcih vinske trte ter osnovne vremenske razmere za določena območja. Razložen je celoten postopek od same ideje za razvoj do končne aplikacije. V prvem poglavju so opisane bolezni in škodljivci vinskih trt ter nadzor nad njimi. Za boljši nadzor je potrebno pridobivanje podatkov, kar je opisano v drugem poglavju. Opisana je vremenska mreža WeatherLink, ki zajema vremenske postaje, s pomočjo katerih pridobivamo podatke, ter način delovanja le-teh. Tretje poglavje zajema zasnovo in tehnični opis rešitve. Našteta in opisana so uporabljena orodja in tehnologije pri razvoju ter sam načrt mobilne aplikacije. V četrtem poglavju je opisano delovanje rešitve, ki zajema izgled in sestavo mobilne aplikacije ter opis pripadajočega API-ja. Na koncu diplomske naloge so podane sklepne ugotovitve ob razvoju mobilne aplikacije in ideje za nadaljnji razvoj.

Ključne besede: mobilna aplikacija, bolezni in škodljivci vinske trte, Vitis.

Abstract

Title: Development of mobile application Vitis

Author: Luka Zlatečan

The diploma work explains the development of the mobile application Vitis. The application will display information about diseases and pests of the vine and the basic weather conditions for certain areas. The whole process is explained from the idea for development to the final application. The first chapter describes the diseases and pests of vines and control over them. For better control, data acquisition is required, as described in Chapter 2, which contains WeatherLink weather network description and the description of the used weather stations and the way they operate. The third chapter covers the design and technical description of the solution. It contains the description of the tools and technologies used in development and the mobile application plan itself. Chapter 4 describes the layout and composition of a mobile application and a description of the corresponding API. The last chapter includes conclusions made while developing the mobile application and the idea for further development.

Keywords: mobile application, diseases and pests of the vine, Vitis.

Poglavje 1

Uvod

Vinogradništvo je na območju Slovenije prisotno že vrsto let in ima za prebivalce zelo velik kulturni in gospodarski pomen. V Sloveniji je bilo po podatkih kmetijsko gozdarske zbornice Slovenije leta 2010 v register pridelovalcev grozdja in vina vpisanih kar 28000 pridelovalcev grozdja in vina. A tako kot večina drugih rastlin je tudi vinska trta izpostavljena različnim boleznim in škodljivcem. V Sloveniji jih poznamo kar nekaj, ki vsako leto povzročijo ogromno škode. Opisi le-teh so povzeti po [12].

1.1 Glivične bolezni vinske trte

Glive so najboljše skupina povzročiteljev bolezni rastlin, ki pričnejo po okužbi oblikovati vegetativen splet nitk (hif). Te se nadalje povezujejo v micelije (podgobje). Po uspešni naselitvi tkiv začnejo oblikovati nespolne trose, ki jih raznašajo veter, dež in žuželke. Ko trosi vzkaliijo na površju organov trte, se prične razvoj novega micelija. Nekatere glive ga oblikujejo le na živih tkivih, druge pa se razvijajo tudi na odmrlih delih trte. Prvim pravimo obligatne zajedalske glive (npr. peronospora in pepelovka), drugim pa fakultativne zajedalske glive (npr. siva plesen in črna pegavost). Znaki okužb pri glivičnih boleznih so enostavno razpoznavni, saj so na okuženih organih skoraj vedno opazne prevleke micelijev in trosonoscev s trosi. Vrsto

glive se določi s pregledom organov pod mikroskopom ter gojenjem na petrijevkah. Najpogostejši znaki okužbe z glivami so pege, plesnive prevleke in gnitja. V Sloveniji so aktivne:

- antraknoza vinske trte - črni pikec (*Elsinoë ampelina* Shear)
- črna grozdna gniloba (*Guignardia bidwelli*)
- črna pegavost vinske trte (*Phomopsis viticola*)
- oidij vinske trte (*Uncinula necator*)
- peronospora vinske trte (*Plasmopara viticola*)
- siva plesen - grozdna gniloba (*Botryotinia fuckeliana*)

1.2 Škodljivci vinske trte

V Sloveniji imamo ogromno škodljivcev vinske trte, vendar pa so le redki med njimi škodljivi za gospodarstvo. Škodljivci objedajo ali izsesavajo nadzemne ali podzemne dele trte. Povzročajo lahko neposredno in posredno škodo. Med posredno škodo uvrščamo okužbe z glivičnimi, virusnimi in drugimi boleznimi. V večini imajo škodljivci več generacij letno. Za območje Slovenije je najbolj značilen pasasti in križasti grozdni sukač (*Eupoecilia ambiguella* in *Lobesia botrana*).

1.3 Nadzor

Stopnje nevarnosti za razvoj škodljivcev in bolezni vinske trte se v zadnjih letih zaradi klimatskih sprememb samo še povečujejo in jih je vedno težje nadzorovati. S pomočjo novih dognanj s področja fitofarmacije poskušajo vinogradniki bolezni in škodljivce vinske trte čim bolj zatreti. Vendar pa je pretirana in napačna uporaba pesticidov tudi škodljiva in za vinogradnike nemalokrat predstavlja velik finančni vložek. Številne univerze in podjetja po svetu so se zavedali tega problema in so s pomočjo preizkusov in testiranj razvili matematične modele, ki na podlagi različnih meritev izračunajo stopnjo nevarnosti za razvoj bolezni in škodljivcev. Vendar pa so ti modeli največkrat za preprostega uporabnika nedostopni ali pa prezahtevni. V sodelovanju z inštitutom IPRA smo se zato odločili, da razvijemo aplikacijo, s pomočjo katere bi informacije o boleznih in škodljivcih lahko poiskal prav vsak. Cilj moje diplomske naloge je bila izdelava mobilne aplikacije Vitis, ki uporabnike opozarja o nevarnostih razvoja bolezni in škodljivcev vinske trte ter jih informira o njih. S pomočjo aplikacije bodo uporabniki lahko spremljali stopnjo nevarnosti za razvoj določenih bolezni trte, ki so pri nas najbolj pogoste. Poleg tega pa bodo na voljo tudi osnovni podatki o škodljivcih, zgozdovina podatkov za posameznega škodljivca in osnovni vremenski podatki za območje meritev.

Poglavje 2

Pridobivanje podatkov

Eno izmed podjetij, ki se ukvarja s problematiko nadzora nad boleznimi in škodljivci rastlin, je Davis, ki se med drugim ukvarja z merilnimi napravami za spremljanje vremenskih razmer. Poleg strojne opreme razvijajo tudi programsko opremo za lažji nadzor in učinkovito izkoriščenost vremenskih podatkov, ki se izmerijo s pomočjo njihovih vremenskih postaj. Med programsko opremo spadajo tudi njihovi IPM moduli, ki so podrobneje opisani v nadaljevanju poglavja. Na žalost pa podatki, pridobljeni s pomočjo programske opreme IPM modulov, niso dostopni širši javnosti in tudi posameznik, ki je uporabnik njihove programske opreme, za večino informacij potrebuje dostop do specifičnega računalnika, ki je vnaprej povezan s postajo. Poleg tega pa je programska oprema precej kompleksna. Glavna ideja aplikacije Vitis pa je prav to, da so podatki dostopni vsem in da za njihovo razumevanje in interpretacijo ni potrebno ogromno tehnološkega znanja.

2.1 WeatherLink

WeatherLink je globalna vremenska mreža podjetja Davis, ki zajema več tisoč vremenskih postaj. Vsaka postaja, vključena v mrežo, je prikazana na spletni strani weatherlink.com, kjer lahko spremljamo vremenske podatke, ki jih zajema postaja. Del mreže teh postaj so tudi postaje, ki so del aplikacije

Vitis. V mrežo postaj se sicer lahko vključijo vremenske postaje različnih tipov, vendar so pri aplikaciji uporabne vremenske postaje tipa Vantage Pro in Vantage Pro 2, ki so kompatibilne z IPM moduli. Postaja Vantage Pro 2 je prikazana na spodnji sliki. S števkami so označeni deli in senzorji postaje.

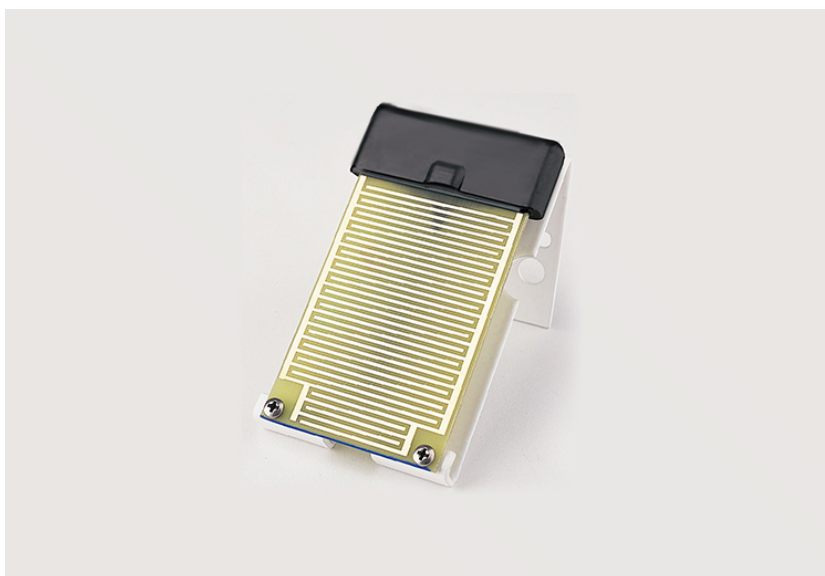


Slika 2.1: Davis vremenska postaja Vantage Pro 2

- | | |
|--------------------|----------------------------|
| 1. Smer vetra | 4. Raven mehurčkov |
| 2. Hitrost vetra | 5. Ščit pred sevanjem |
| 3. Zbiralnik dežja | 6. Temperatura in vlažnost |

- | | |
|-----------------------------|--------------------------|
| 7. Sevanje in UV svetloba | 10. Sončna celica |
| 8. Brezžično/kabelsko | |
| 9. Vremensko odporno ohišje | 11. Prilagodljivo držalo |

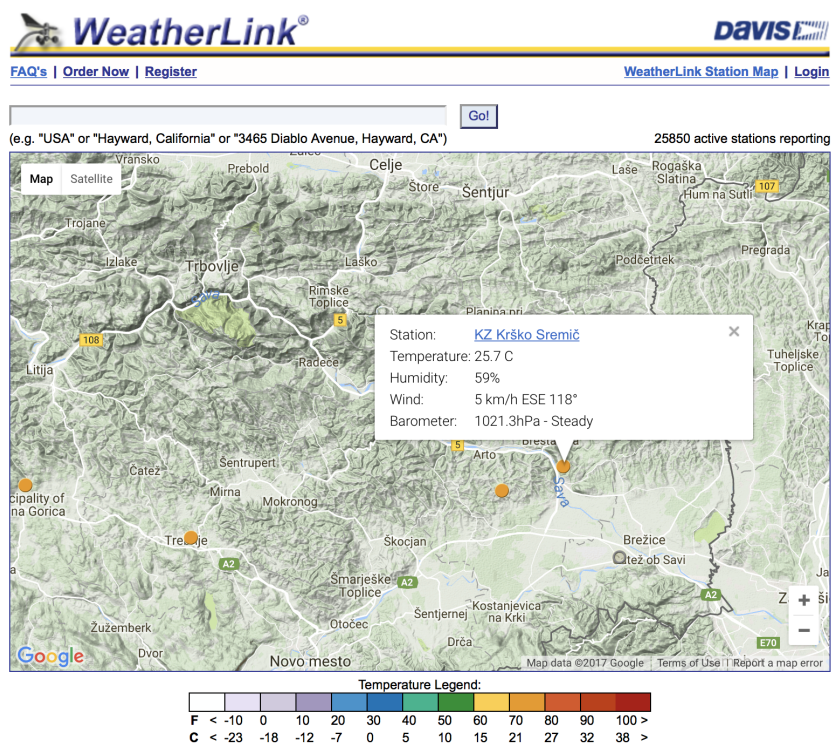
Poleg standardnih delov vremenske postaje je možno priključiti tudi dodatne senzorje. V primeru razvoja bolezni in škodljivcev vinske trte je zelo pomembna omočenost listov, zato je potrebno na vremensko postajo priključiti tudi senzor za omočenost listov, ki omogoča spremljanje omočenosti površine na skali med 0, ki pomeni popolnoma suho, in 15, ki pomeni nasičeno mokro.



Slika 2.2: Senzor za omočenost listov

Postaje so lahko povezane preko kabla ali pa brezžično. Dostop do podatkov vremenske postaje in njihova analiza je uporabniku omogočena s pomočjo produkta WeatherLinkIP. Ta vsebuje programsko opremo WeatherLink in podatkovni zapisovalnik, ki ga povežemo s konzolo vremenske postaje. Ta zapisovalnik prenaša podatke, ki jih je vremenska postaja zabeležila, na računalnik, ki je vnaprej konfiguriran za določeno postajo. Tako si ustvarjamo lokalno bazo vremenskih podatkov. Podatke lahko nato pregledujemo

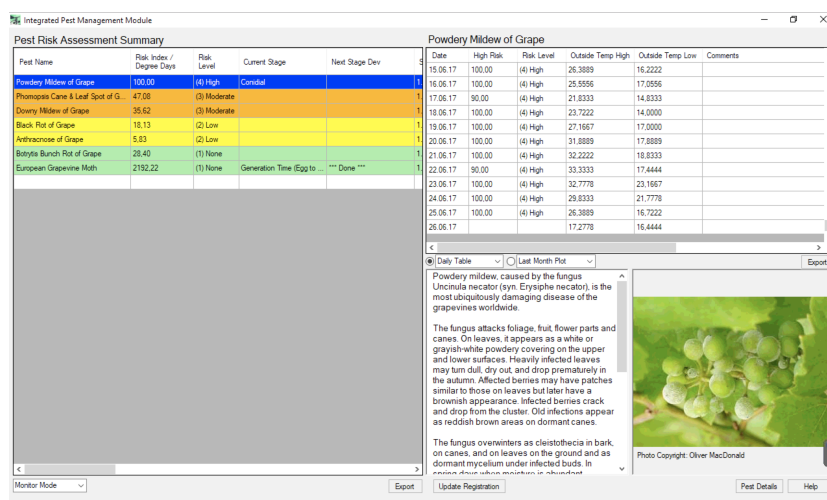
in analiziramo s pomočjo njihovega programa WeatherLink. Poleg prenosa na računalnik, podatkovni zapisovalnik prenaša podatke tudi na spletni strežnik, ki ga ponuja Davis, tako da so podatki dostopni tudi preko spleta na njihovi spletni strani. [9]



Slika 2.3: Spletna stran WeatherLink

2.2 IPM modul za grozdje

K standardni opremi je na voljo tudi več strojnih in programskih dodatkov. Med programske dodatke spadajo tudi IPM moduli podjetja Davis, ki služijo kot dodatek k programski opremi WeatherLink in služijo za učinkovito informiranje uporabnikov o preventivni zaščiti pridelka. Trenutno so na voljo IPM moduli za grozdje, jabolka in hruške, koščičnice ter oreške. Za potrebe naše aplikacije potrebujemo samo IPM modul za grozdje. Ta zbira podatke o lokalni temperaturi, vlažnosti, dežju in omočenosti listov ter na podlagi le-teh s pomočjo različnih modelov izračuna stopnjo nevarnosti za razvoj določenih bolezni. Modul lahko poljubno prilagodimo za spremljanje bolezni in insektov, ki so aktivni in značilni za naše območje. S pomočjo izračunanih nevarnosti lahko hitro ugotovimo, kdaj so potrebni ukrepi za nadzor škodljivcev. IPM modul je, tako kot vremenski podatki, dosegljiv tudi preko spleta, vendar pa trenutno vključene postaje v našo aplikacijo še nimajo te funkcionalnosti, zato se podatki o boleznih in insektih črpajo lokalno preko računalnikov. [6]

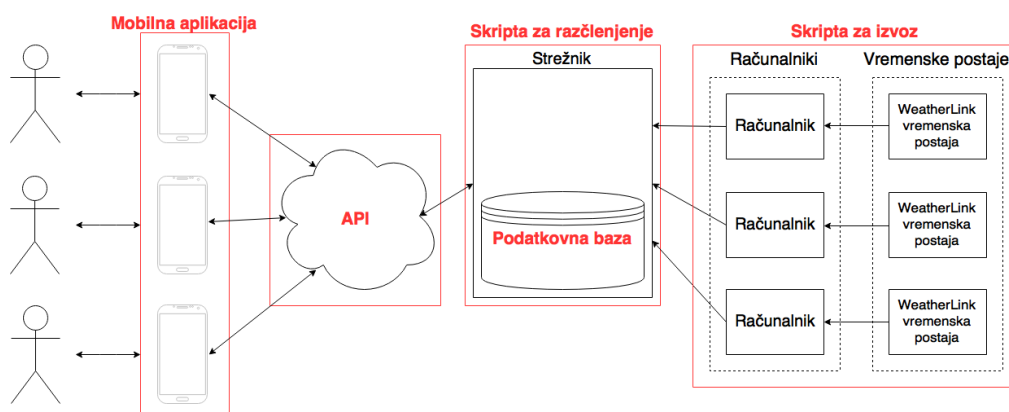


Slika 2.4: Okno IPM modula za grozdje

Poglavje 3

Zasnova in tehnični opis rešitve

Želel sem razviti mobilno aplikacijo, ki bi bila dostopna na vseh mobilnih platformah ter bi prikazovala podatke o nevarnostih za razvoj bolezni in škodljivcev vinske trte. Aplikacija naj bo razdeljena na pet komponent. Prva komponenta aplikacije naj bo skripta za izvoz podatkov, ki skrbi, da se na računalnikih vsako uro zažene program WeatherLink in dodatek IPM modul ter izvozi podatke na centralni strežnik. Druga komponenta aplikacije naj bo skripta za razčlenjenje podatkov, ki vsako uro preveri, ali so bile na centralni strežnik odložene kakšne datoteke ter jih ustrezno prebere in podatke zapiše v podatkovno bazo, ki naj bo tretja komponenta aplikacije. Četrta komponenta aplikacije naj bo API, ki omogoča mobilni aplikaciji prejemanje podatkov v obliki JSON. Peta komponenta naj bo mobilna aplikacija. Komponente so prikazane na naslednji sliki.



Slika 3.1: Komponente aplikacije

3.1 Skripta za izvoz


Vremenske postaje so sinhronizirane vsaka s svojim računalnikom. Nanje vsake nekaj minut pošljejo vremenske podatke, ki jih odčitajo s pomočjo nameščenih senzorjev. Trenutno je na vremenskih postajah časovni interval osvežitve podatkov nastavljen na 10 minut, vendar pa se da le-ta prilagajati. Vsak izmed računalnikov prejme podatke in jih shrani v lokalne datoteke. Na računalnikih, ki so namenjeni vsak svoji vremenski postaji, mora biti nameščena powershell skripta, ki se s pomočjo Windows Task Scheduler zažene vsako uro. Skripta za izvoz zažene program WeatherLink in znotraj njega IPM modul, ki na podlagi zgodovine vremenskih podatkov na postaji, kot je opisano v poglavju o vremenskih postajah, izračuna stopnjo nevarnosti za posamezne boleznitvorne trte. Ti podatki se nato izvozijo v CSV datoteko, ki naj ima ime [n].csv, pri čemer je n ID postaje, kot je vnaprej določeno. S pomočjo programa Puty se morajo datoteke izvoziti v dogovorjeno mapo na centralnem strežniku, na katerem so zbrani podatki o izračunanih nevarnostih za vse postaje, ki jih želimo vključiti.

3.2 Skripta za razčlenjenje

Na centralnem strežniku mora biti nameščen cron job, ki vsako uro zažene skripto za razčlenjenje, ki preveri, ali je bila v vnaprej določeno mapo na sistemu odložena kakšna CSV datoteka. Spodaj je primer cron datoteke, ki skrbi, da se vsako uro zaženeta skripti `checkAndImport.sh` in `getStationsData.sh`.

```
# m h dom mon dow  command
0 * * * * /home/luka/vitis-api/cron/checkAndImport.sh
0 * * * * /home/luka/vitis-api/cron/getStationsData.sh
```

CSV datoteka se naj nato s pomočjo bash skripte razčleni in zapiše v podatkovno bazo. Nekateri podatki se morajo prevesti oziroma prilagoditi potrebam aplikacije. Struktura datotek je fiksna, zato z razčlenitvijo in prilagoditvami ne sme biti težav. Na sliki je primer CSV datoteke, ki je bila odložena na centralni strežnik in jo je potrebno razčleniti ter podatke zapisati v podatkovno bazo.



Pest Name	Risk Index	Degree Days	Risk Level	Current Stage	Next Stage	Dev	Start Date	Last Action Date	Temp Sensor	Hum Sensor	Leaf Wet Sensor
Powdery Mildew of Grape	90,00	(4)	High	Conidial			1.1.2012	1.1.2012	Grapes	Disease	21,11,29,44,35,00,,,8
Downy Mildew of Grape	65,57	(4)	High				1.1.2012	1.1.2012	Grapes	Disease	10,00,,33,00,70,93,8,3,12,40,,,1
Phomopsis Cane & Leaf Spot of Grape Seyval	57,98	(3)	Moderate				1.1.2012	1.1.2012	Grapes	Disease	5,00,,,3
Black Rot of Grape	29,15	(3)	Moderate				1.1.2012	1.1.2012	Grapes	Disease	7,00,26,50,33,00,,,8,5,20,50,,,
Anthracoise of Grape	11,47	(2)	Low				1.1.2012	1.1.2012	Grapes	Disease	2,00,26,00,33,00,,,8,4,15,50,,,kr
European Grapevine Moth	4490,23	(1)	None	Generation Time (Egg to Adult): Restart Model on Biofix Date,							
Botrytis Bunch Rot of Grape	46,99	(1)	None				1.1.2012	1.1.2012	Grapes	Disease	12,00,32,00,40,00,,,8,50,6

Slika 3.2: Primer CSV datoteke

3.3 Podatkovna baza

Za hranjenje podatkov potrebujemo štiri tabele. Prva tabela hrani splošne podatke o boleznih in škodljivcih vinske trte in jo poimenujemo PESTS. Druga tabela hrani splošne podatke o vključenih vremenskih postajah in jo poimenujemo STATIONS. Tretja tabela hrani podatke o vremenskih razmerah za vse vremenske postaje in jo poimenujemo STATIONDATA. Četrta

tabela hrani podatke o stopnjah nevarnosti za razvoj bolezni in škodljivcev vinske trte za vse postaje (shranjene v tabeli STATIONS) in vse škodljivce in bolezni (shranjene v tabeli PESTS) ter jo poimenujemo WARNINGS. Spodaj so podani podrobnejši opisi podatkovnih tabel, ki jih potrebujemo, in podatkovni model celotne podatkovne baze.

3.3.1 Tabela PESTS

V tabeli PESTS so shranjeni podatki o boleznih in škodljivcih vinske trte, ki jih aplikacija zajema. Vsaka bolezen oziroma škodljivec ima svoj ID, tip, ime, latinsko ime, opis, ime datoteke od slike v določeni mapi in datum vnosa v podatkovno bazo. Škodljivcev in bolezni je stalno število in se bodo v prihodnosti redko dodajali, zato je potrebno vsakega izmed njih dodati z INSERT ukazom.

Ime stolpca	Tip spremenljivke	Not null	Komentar
PESTID	INT(32) UNSIGNED	X	Unique ID for each pest
PESTTYPE	VARCHAR(64)	X	Bolezen, Insekt
PESTNAME	VARCHAR(128)	X	Name of pest
PESTNAMELATIN	VARCHAR(128)	X	Latin name of pest
PESTDDESCRIPTION	LONGTEXT		Description
PESTPATH	VARCHAR(256)		Path to image of pest
PESTDATCRE	DATETIME	X	Creation date

Tabela 3.2: Tabela PESTS

3.3.2 Tabela STATIONS

V tabeli STATIONS so shranjeni podatki o vremenskih postajah. Vsaka postaja ima svoj ID, ime, datum vnosa v podatkovno bazo, geografsko širino, geografsko dolžino, nadmorsko višino in višino, na kateri je postaja postavljena, merjeno od nadmorske višine navzgor. Število postaj se trenutno še ne spreminja tako hitro, da bi potrebovali vnosno masko, zato je tako kot pri boleznih in škodljivcih potrebno vsako izmed postaj dodati ročno s pomočjo INSERT ukaza.

Ime stolpca	Tip spremenljivke	Not null	Komentar
STATID	INT(32) UNSIGNED	X	Unique ID for each station
STATNAME	VARCHAR(128)	X	Name of station
STATDATCRE	DATETIME	X	Creation date
STATGEOLAT	FLOAT(32) UNSIGNED	X	Geo latitude of station
STATGEOLNG	FLOAT(32) UNSIGNED	X	Geo longitude of station
STATGEOALT	FLOAT(32) UNSIGNED	X	Geo altitude of station
STATHEIGHT	FLOAT(32) UNSIGNED	X	Height of station

Tabela 3.4: Tabela STATIONS

3.3.3 Tabela STATIONDATA

V tabeli STATIONDATA so shranjeni podatki, zajeti iz vremenske postaje. Vsaka vrstica v tabeli predstavlja vremensko stanje določene postaje v določenem trenutku. Vsaka vrstica ima svoj ID, ID postaje, temperaturo, vlažnost, dež, tlak in datum vnosa v podatkovno bazo. Podatki se v tabelo zapišejo vsako polno uro s pomočjo bash skripte, ki razčleni podatke s spletne strani WeatherLink, ki zbira podatke od vseh njihovih postaj.

Ime stolpca	Tip spremenljivke	Not null	Komentar
STDATTEMPID	INT(64) UNSIGNED	X	Unique ID for each data
STDATSTATID	INT(32) UNSIGNED	X	ID of station
STDATTEMP	FLOAT(32)	X	Temperature for station
STDATHUM	FLOAT(32)	X	Humidity for station
STDATRAIN	FLOAT(32)	X	Rain for station
STDATBAR	FLOAT(32)	X	Barometer for station
STDATDATCRE	DATETIME	X	Creation date

Tabela 3.6: Tabela STATIONDATA

3.3.4 Tabela WARNINGS

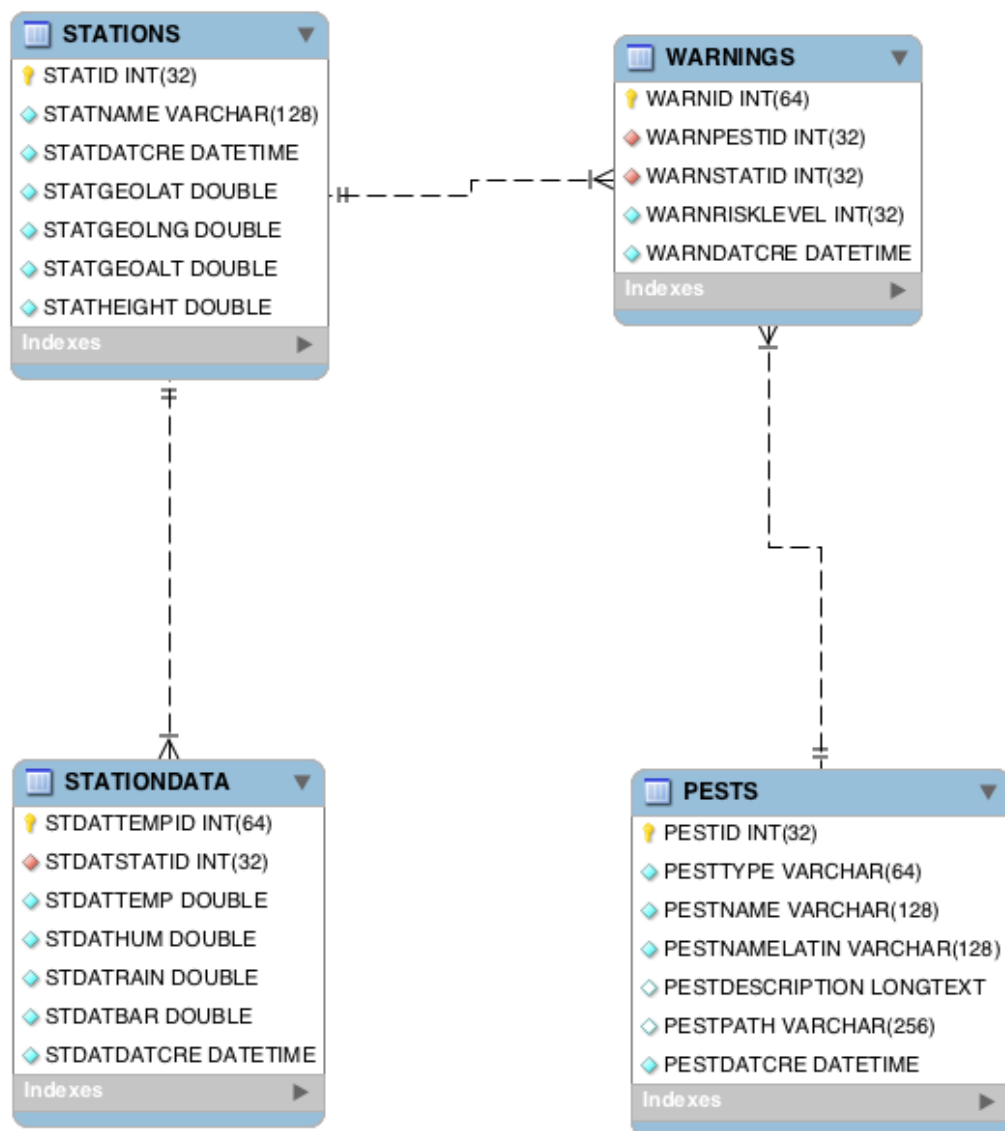
V tabeli WARNINGS so shranjeni podatki o trenutnih stopnjah nevarnosti za vsako bolezen oziroma škodljivca za vsako vremensko postajo. Vsaka vrstica ima svoj ID, ID škodljivca oziroma bolezni, ID postaje, stopnjo nevarnosti in datum vnosa v podatkovno bazo. Podatki se v tabelo zapišejo vsako polno uro s pomočjo bash skripte za razčlenitev, ki razčleni CSV datoteko, ki jo računalniki, sinhronizirani z vremenskimi postajami, odložijo na vnaprej določeno lokacijo na sistemu.

Ime stolpca	Tip spremenljivke	Not null	Komentar
WARNID	INT(64) UNSIGNED	X	Unique ID for each warning
WARNPESTID	INT(32) UNSIGNED	X	ID of pest
WARNSTATID	INT(32) UNSIGNED	X	ID of station
WARNRISKLEVEL	INT(32) UNSIGNED	X	Risk level for pest
WARNDATCRE	DATETIME	X	Creation date

Tabela 3.8: Tabela WARNINGS

3.3.5 Podatkovni model

Vse skupaj najlažje prikažemo s podatkovnim modelom.



Slika 3.3: Podatkovni model

3.4 API

API mora biti preprost za uporabo in mora podpirati vse funkcionalnosti, ki jih bo mobilna aplikacija potrebovala za delovanje. Zgrajen je v REST arhitekturi in za strežnik uporablja NodeJS. Kot NodeJS okvir je najlažje uporabiti express, ki je namenjen predvsem razvoju spletnih aplikacij. Za povezavo na mySQL podatkovno bazo je priporočljiv modul „mysql“. API mora omogočati samo GET HTTP metodo, saj bo uporabljen samo za pridobivanje podatkov. Za lažje razumevanje delovanja API-ja je treba ustvariti tudi ustrezno dokumentacijo. Za to je najlažje uporabiti kar APIDOC, ki s pomočjo komentarjev znotraj izvirne kode ustvari uporabniku prijazno spletno stran z vsemi informacijami o API-ju. API mora omogočati klice, ki bodo vračali sezname in podrobnosti o boleznih in škodljivcih vinske trte, vremenskih postajah, stopnjah nevarnosti za razvoj bolezni in škodljivcev ter vremenske razmere.

3.5 Mobilna aplikacija

Mobilna aplikacija mora tako kot vse imeti svoj logo in začetni zaslon. Logo mora predstavljati vsebino aplikacije, zato je najbolj primeren logo tak, ki predstavlja bolezen vinskih trt. Aplikacija mora biti preprosta in uporabniku prijazna. Vsebovati mora seznam vseh bolezni in škodljivcev vinske trte ter njihove lastnosti in stopnje nevarnosti za določene vremenske postaje. Za vsako bolezen ali škodljivca pa je uporabno, da je na voljo tudi zgodovina podatkov za posamezne vremenske postaje. Vremenske postaje naj bodo predstavljene tudi na zemljevidu, tako da bo uporabnik vedel, za katero območje so podatki iz določene postaje. Za vsako izmed vremenskih postaj pa naj bodo dodane vremenske razmere. Aplikacija mora podatke črpati preko klicev na API, omenjen v prejšnjem odseku.

3.6 Uporabljen orodja in tehnologije pri razvoju

3.6.1 Windows PowerShell

Windows PowerShell je skriptni jezik, zasnovan predvsem za sistemsko administracijo. Zgrajen je na .NET okvirju in je namenjen za pomoč pri nadzoru vanju in avtomatizaciji upravljanja sistema Windows in njegovih aplikacij. V primeru aplikacije Vitis je Windows PowerShell uporabljen za namen zagona programa WeatherLink ter IPM modula ter za zagon programa Putty, ki s pomočjo ukaza SCP izvozi CSV datoteke na centralni strežnik. [7]

3.6.2 Crontab

Crontab (CRON TABLE) je dodatek, ki uporabniku unix sistemov omogoča avtomatizacijo nalog, ki se izvajajo v ozadju. Naloge oziroma ukaze se zapisuje v datoteko crontab. Določa se lahko intervale izvajanja ukazov po specifični sintaksi. Naslednja vrstica v crontab datoteki bi pomenila, da se vsak dan vsakega meseca ob 12:00 izvrši ukaz ping.

```
0 12 * * * ping www.google.si
```

3.6.3 Ionic

Ionic je odprtokodni SDK, ki omogoča razvoj zmogljivih in kvalitetnih mobilnih aplikacij s pomočjo spletnih tehnologij, kot so HTML, CSS in JavaScript. Novembra 2013 je izšla alpha verzija, ki so jo razvili Ben Sperry, Adam Bradley in Max Lynch. Do leta 2015 je bilo s pomočjo Ionica ustvarjenih preko 1,3 milijona aplikacij. Namen Ionica je predvsem izboljšati izgled aplikacije in izboljšati uporabniško izkušnjo. Trenutno je Ionic najboljše uporabljati v kombinaciji z AngularJS, saj lahko z njegovo pomočjo ustvarjamo dinamične aplikacije, ki so uporabniku prijazne in zanimive. Ionic ponuja tudi lasten CLI, ki z naborom ukazov olajša delo razvijalcem mobilnih aplikacij. CLI pa ponuja tudi vgrajeni razvijalski strežnik, orodja za grajenje in razhroščevanje aplikacij. Za tiste, ki pa uporabljajo storitev Ionic Cloud, pa je CLI lahko uporaben tudi kot orodje za izvoz aplikacije. [5]

3.6.4 AngularJS

AngularJS je razširitveni okvir za HTML jezik. Omogoča lažje in bolj dinamično izražanje komponent dinamičnih spletnih aplikacij. [3]

3.6.5 MySQL

MySQL je najbolj popularen odprtokodni SQL sistem. Razvijajo ga pri podjetju Oracle Corporation. MySQL podatkovna baza je relacijska, kar pomeni, da podatke shranjuje v različne tabele, namesto da bi jih shranjevala v eno veliko skladišče podatkov. MySQL za delovanje ne potrebuje ekstremno zmogljivih računalnikov, zato ga lahko uporablja prav vsak na svojem računalniku, ne da bi sploh opazil. Njegova podatkovna programska oprema deluje po principu strežnika, ki podpira različne uporabniške programe in knjižnice ter ogromno API-jev. [10]

3.6.6 Node JS

NodeJS je odprtokodni strežniški okvir, ki teče na več platformah. Namenjen je predvsem gradnji omrežnih aplikacij. Razvil ga je Ryan Dahl leta 2009. NodeJS uporablja asinhroni, pretočni vhodno-izhodni model, ki ne blokira hkratnih zahtev, kar ga dela zelo hitrega in učinkovitega. Primeren je predvsem za aplikacije, ki zahtevajo veliko podatkov v realnem času na več napravah naenkrat. [2, 11]

3.6.7 Git

Git je VCS sistem, ki za razliko od ostalih VCS sistemov, kot so CVS, Subversion, Perforce, Bazar in ostali, ki informacije shranjujejo kot niz datotek in spremembe v vsaki izmed teh datotek v daljšem časovnem obdobju, gleda na podatke kot posnetke datotečnega sistema. Vsakič, ko se shrani stanje projekta, se v bistvu posname slika o trenutnem stanju datotek in kasneje se potem lahko sklicujemo na ta posnetek. Git nespremenjenih datotek ne obravnava ponovno in ne dela novega posnetka, ampak le povezavo do prejšnje enake datoteke, ki jo je posnel. [1]

3.6.8 Sublime Text 3

Sublime Text je preprost, a kljub temu zelo zmogljiv urejevalnik besedil, namenjen predvsem programiranju. Na voljo ima prepoznavo sintaks najrazličnejših programskih jezikov, kar ga dela uporabniku zelo prijaznega. Za prenos je dosegljiv uporabnikom skoraj vseh platform. Poleg vsega pa je možno dodajati lastne vtičnike, ki programiranje še olajšajo. [8]

3.6.9 APIDOC

APIDOC je orodje, ki ustvari dokumentacijo s pomočjo komentarjev znotraj naše kode, ki jih moramo napisati s predpisano sintakso. Naslednji večvrstični komentar prikazuje primer komentarja znotraj kode. [4]

```
/**
 * @api {get} /stations/:id Request station information
 * @apiName GetStation
 * @apiGroup Stationm
 *
 * @apiParam {Number} id Stations unique ID.
 *
 * @apiSuccess {String} name Name of the Station.
 */
```


Poglavje 4

Opis delovanja rešitve

Razvil sem mobilno aplikacijo Vitis, ki je na voljo za vse mobilne platforme in je trenutno dostopna na naslovu `krsko.ipra.si`. Vzporedno z razvojem mobilne aplikacije sem razvil tudi API, preko katerega aplikacija pridobiva podatke.

4.1 API

API je zgrajen s pomočjo NodeJS okvirja „express“ in modula „mysql“ za povezavo na podatkovno bazo. API omogoča uporabniku oziroma programerju osem različnih HTTP GET zahtev, s katerimi pridobiva informacije o boleznih, vremenskih postajah, škodljivcih in boleznih vinske trte ter drugih stvareh. Vseh osem je predstavljenih v naslednjih podpoglavjih. Format odgovora na API klice je JSON, razen v primeru slik. Spodaj so podrobno opisane vse podprte funkcionalnosti API-ja. Vsaka izmed funkcionalnosti je znotraj izvirne kode ustrezno komentirana, tako da je pregled API-ja dosegljiv tudi na spletni strani, ki se sama generira s pomočjo orodja APIDOC glede na komentarje znotraj izvirne kode. Spodaj je primer izseka spletne strani, ustvarjene s pomočjo orodja APIDOC.

Pests

Pests - Request list of all pests 0.1.0 -

GET

`http://krsko.ipra.si/v1/pests`

Success 200

Field	Type	Description
id	Integer	ID of the pest.
name	String	Name of the pest.

Success-Response:

```
HTTP/1.1 200 OK
[
  {
    "id": 1,
    "name": "Antraknoza vinske trte"
  },
  {
    "id": 2,
    "name": "Črna grozdna gniloba"
  },
  {
    "id": 3,
    "name": "Siva grozdna plesena"
  },
  {
    "id": 4,
    "name": "Peronospora"
  },
  {
    "id": 5,
    "name": "Črna pegavost vinske trte"
  },
  {
    "id": 6,
    "name": "Oidij vinske trte"
  }
]
```

Slika 4.1: Spletna stran, generirana s pomočjo orodja APIDOC

4.1.1 Seznam vremenskih postaj

Vir: /stations

Parametri: /

Pomen: Seznam vseh vključenih vremenskih postaj z njihovimi podrobnimi geografskimi podatki

Primer odgovora na klic:

```
[
  {
    "id": 1,
    "geoLatitude": 45.97333333333336,
    "geoLongitude": 15.493611111111111,
    "geoAltitude": 336.1,
    "height": 3,
    "name": "KZ Krško Sremič"
  },
  {
    "id": 2,
    "geoLatitude": 45.95194444444445,
    "geoLongitude": 15.415277777777778,
    "geoAltitude": 363.4,
    "height": 6,
    "name": "KZ Krško Vrhulje"
  }
]
```

4.1.2 Podrobnosti vremenske postaje

Vir: /stations/:id

Parametri:

- id - ID vremenske postaje

Pomen: Podrobni geografski podatki o vremenski postaji z željenim ID-jem

Primer odgovora na klic:

```
{
  "id": 1,
  "name": "KZ Krško Sremič",
  "geoLatitude": 45.97333333333336,
  "geoLongitude": 15.493611111111111,
  "geoAltitude": 336.1,
  "height": 3
}
```

4.1.3 Vremenske razmere

Vir: /stations/:id/weather

Parametri:

- id - ID vremenske postaje

Pomen: Vremenske razmere na območju postaje z željenim ID-jem

Primer odgovora na klic:

```
{
  "temperature": 24,
  "humidity": 54,
  "rain": 6.4,
  "pressure": 1019.3,
  "lastUpdated": "7.8.2017 14:00"
}
```

4.1.4 Trenutne stopnje nevarnosti za razvoj bolezni in škodljivcev

Vir: /stations/:id/warnings

Parametri:

- id - ID vremenske postaje

Pomen: Stopnje nevarnosti za razvoj bolezni in škodljivcev vinske trte za območje vremenske postaje z željenim ID-jem

Primer odgovora na klic:

```
[
  {
    "name": "Oidij vinske trte",
    "riskLevel": 4,
    "lastUpdated": "7/8/2017 15:00"
  },
  {
    "name": "Črna pegavost vinske trte",
    "riskLevel": 2,
    "lastUpdated": "7/8/2017 15:00"
  },
  {
    "name": "Peronospora",
    "riskLevel": 2,
    "lastUpdated": "7/8/2017 15:00"
  },
  ...
]
```

4.1.5 Stopnje nevarnosti za razvoj bolezni in škodljivcev za daljše obdobje

Vir:/stations/:id/warnings/:pestId/:period

Parametri:

- id - ID vremenske postaje
- pestId - ID škodljivca oziroma bolezni
- period - Željeno obdobje („day“, „week“, „month“, „year“)

Pomen: Povprečje stopenj nevarnosti za trenutni dan, teden, mesec, leto za razvoj bolezni in škodljivcev vinske trte z željenim ID-jem za območje vremenske postaje z željenim ID-jem

Primer odgovora na klic:

```
[
  {
    "period": 1,
    "riskLevel": 1
  },
  {
    "period": 2,
    "riskLevel": 1
  },
  {
    "period": 3,
    "riskLevel": 1
  },
  ...
]
```

4.1.6 Bolezni in škodljivci

Vir: /pests

Parametri: /

Pomen: Seznam vseh bolezni in škodljivcev

Primer odgovora na klic:

```
[
  {
    "id": 1,
    "name": "Antraknoza vinske trte"
  },
  {
    "id": 2,
    "name": "Črna grozdna gniloba"
  },
  {
    "id": 3,
    "name": "Siva grozdna plesen"
  },
  ...
]
```


4.1.7 Podrobnosti o boleznih in škodljivcih

Vir: /pests/:id

Parametri:

- id - ID bolezni oziroma škodljivca

Pomen: Podrobnosti škodljivca oziroma bolezni vinske trte z željenim ID-jem

Primer odgovora na klic:

```
{
  "id": 1,
  "name": "Antraknoza vinske trte",
  "nameLatin": "Elsinoe ampelina",
  "description": "Gliva, ki okuži zelene dele rastline, ..."
}
```

4.1.8 Slika bolezni in škodljivcev

Vir: /pests/:id/img

Parametri:

- id - ID bolezni oziroma škodljivca

Pomen: Slika škodljivca oziroma bolezni vinske trte z željenim ID-jem

Primer odgovora na klic:



Slika 4.2: Primer odgovora na klic

4.2 Mobilna aplikacija

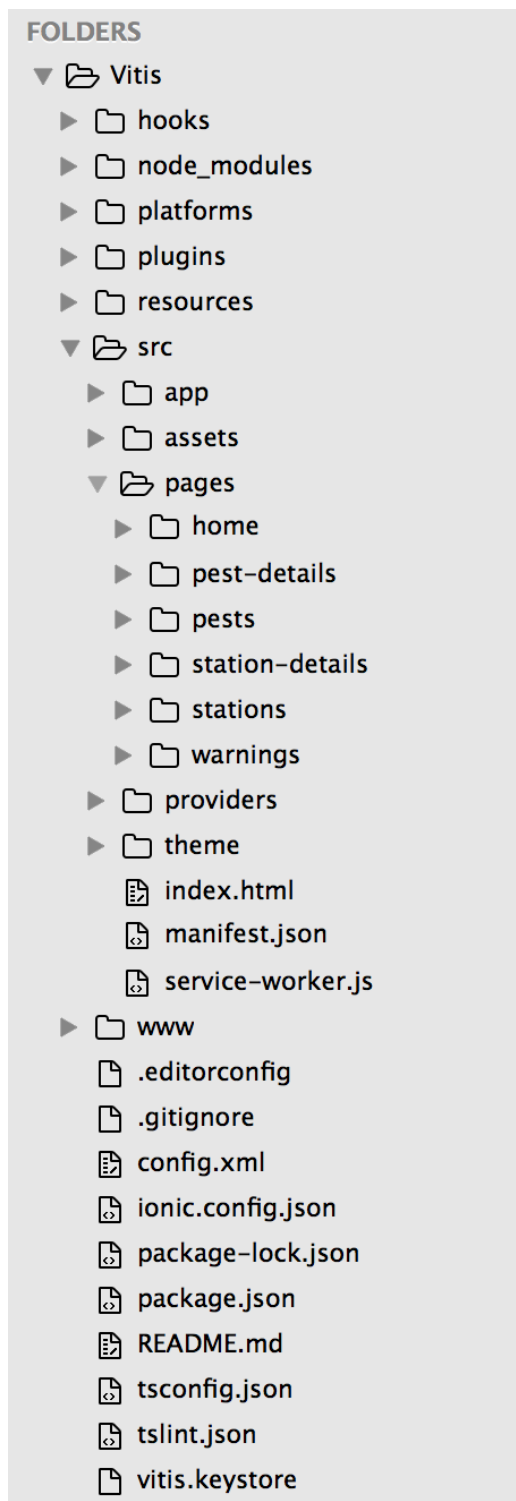
Mobilna aplikacija je napisana s pomočjo Ionic SDK-ja, ki omogoča hitre in enostavne začetke gradnje aplikacije. Ogrodje aplikacije lahko postavimo že s preprostim ukazom:

```
ionic start test blank
```

Vsak tako ustvarjen Ionic projekt vsebuje značilne mape in datoteke. V spodnji tabeli so predstavljene nekatere mape in datoteke.

Mapa/Datoteka	Pomen
node_modules	Vsi moduli, ki jih potrebujemo pri razvoju aplikacije
platforms	Datoteke, specifične za mobilne platforme, ki so dodane v razvoj
plugins	Vtičniki, potrebni za razvoj aplikacije (največ cordova)
resources	Slike, ikone in splash ekran
src	Izvorna koda izgleda in funkcionalnosti aplikacije
www	Zgrajena spletna različica aplikacije, primerna za testiranje na računalniku
config.xml	Konfiguracijska datoteka aplikacije
ionic.config.json	Konfiguracijska datoteka Ionic projekta
package.json	Konfiguracijska datoteka projekta, kjer navedemo potrebne vtičnike in module, ki jih potrebujemo pri razvoju aplikacije

Tabela 4.2: Glavne mape in datoteke Ionic projekta



Slika 4.3: Struktura Ionic projekta

Aplikacijo lahko s pomočjo Ionic CLI hitro zaženemo in testiramo v spletnem brskalniku s pomočjo ukaza:

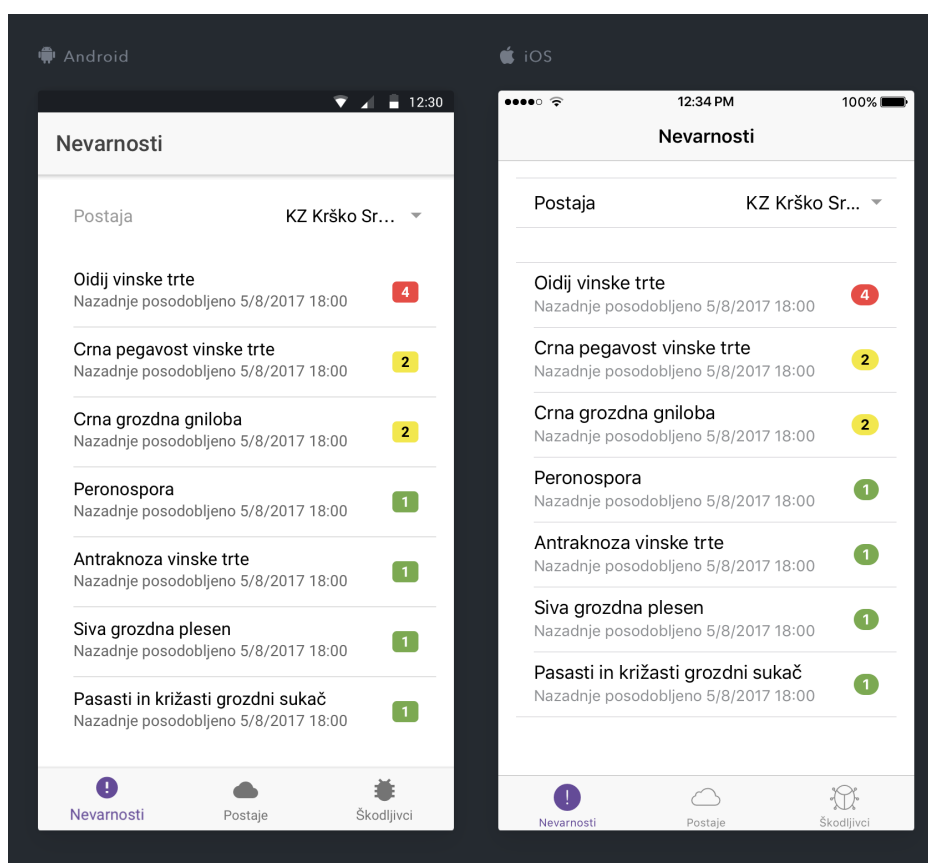
```
ionic serve
```

Poskrbeti moramo, da smo v mapi projekta. Po potrebi lahko za namene razhroščevanja uporabimo tudi stikalo.

```
ionic serve -c
```

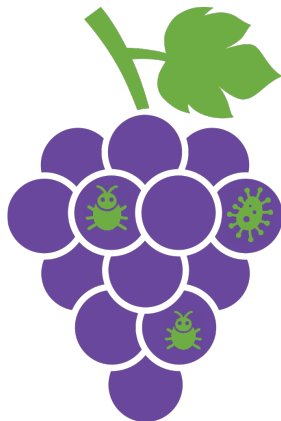
Če pa želimo aplikacijo preizkusiti v mobilnem izgledu, pa uporabimo stikalo za to.

```
ionic serve --lab
```



Slika 4.4: Preizkus mobilne aplikacije s stikalom –lab

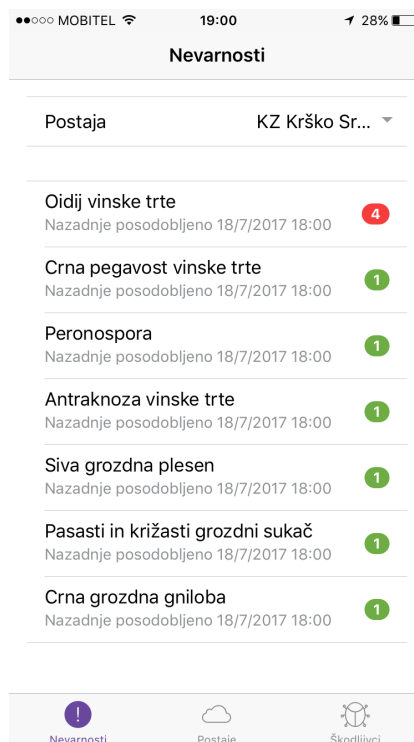
Ob vstopu se odpre glavno okno aplikacije, ki je sestavljeno iz treh zavihkov, ki so Nevarnosti, Postaje in Škodljivci. Aplikacija ima tudi svoj logo, ki je sestavljen iz grozda in škodljivcev. V njem se prepletata dve glavni barvi, ki sta tudi tematski barvi celotne aplikacije.



Slika 4.5: Logo aplikacije

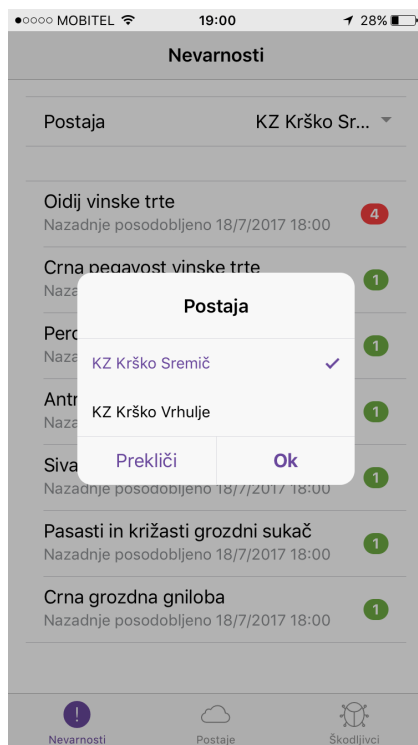
4.2.1 Nevarnosti

Ob zagonu aplikacije je nastavljeno, da je privzeti odprti zavihek Nevarnosti. Na zavihku Nevarnosti lahko uporabnik spremlja stopnje nevarnosti za posamezne bolezni in škodljivce.



Slika 4.6: Zavihek z nevarnostmi

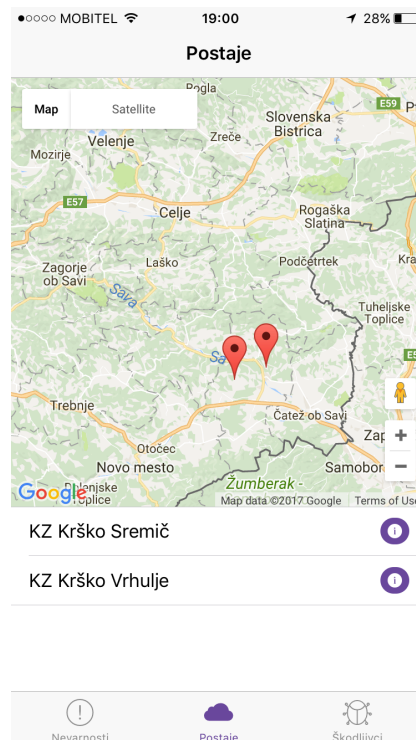
Podatki so prikazani za eno postajo. Uporabnik lahko po želji menja postajo, za katero želi videti stopnje nevarnosti.



Slika 4.7: Izbira postaje v zavihku Nevarnosti

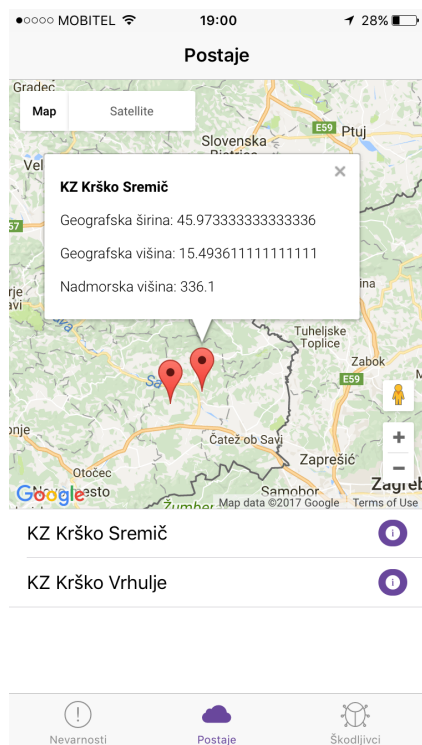
4.2.2 Postaje

Drug zavihek vsebuje seznam vremenskih postaj, ki so vključene v aplikacijo. Za vsako izmed postaj lahko pridobimo tudi podatke o vremenu s pritiskom na gumb za informacije ob postaji. Poleg seznama je prikazan tudi zemljevid z lokacijami vremenskih postaj.



Slika 4.8: Zavihek z vremenskimi postajami

S klikom na postajo na zemljevidu lahko izvemo dodatne geografske informacije o postaji.



Slika 4.9: Geografski podatki o posamezni postaji

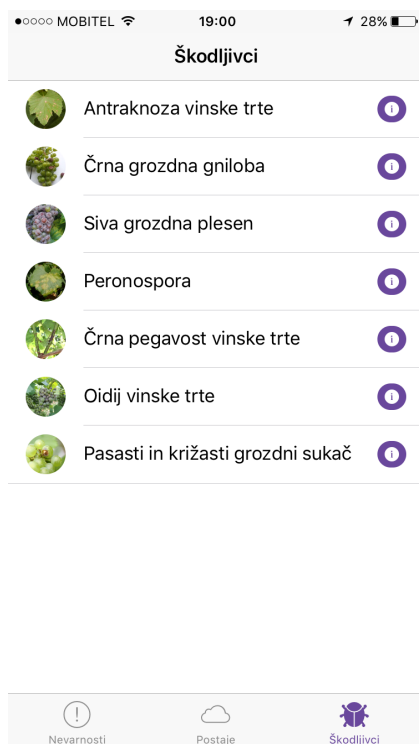
S klikom na gumb za informacije lahko za posamezno postajo vidimo, kakšna je bila tam temperatura, vlažnost, količina padavin in tlak ob zadnji posodobitvi podatkov.



Slika 4.10: Vremenski podatki za posamezno postajo

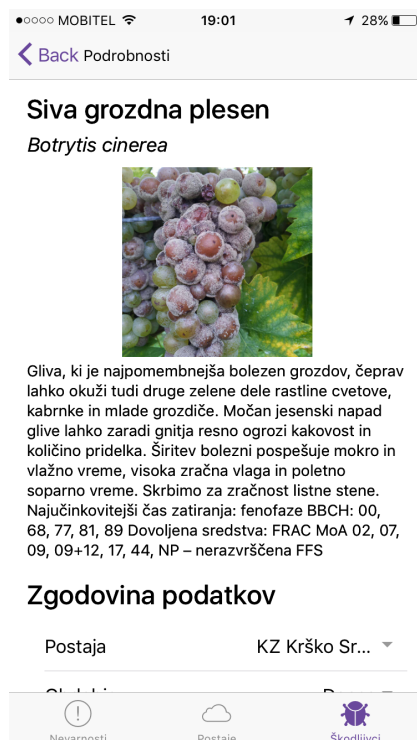
4.2.3 Škodljivci

Tretji zavihek vsebuje seznam vseh bolezni in škodljivcev vinske trte, ki jih aplikacija zajema.



Slika 4.11: Zavihek s škodljivci

O vsakem škodljivcu lahko s klikom na gumb za informacije izvemo več podrobnosti. Za vsakega izmed škodljivcev oziroma bolezni je napisan krajši opis, ki so ga pripravili na inštitutu Ipra. Poleg opisa pa je priložena tudi slika, ki jo škodljivec pusti na vinski trti.



Slika 4.12: Podrobnosti o posameznem škodljivcu

Na dnu strani je graf, ki prikazuje zgodovino nevarnosti za razvoj določene bolezni oziroma škodljivcev. Na grafu lahko izbiramo med postajami ter periodami, za katere želimo videti zgodovino podatkov.



Slika 4.13: Grafična ponazoritev nevarnosti za razvoj škodljivca

Poglavje 5

Sklep

Aplikacija Vitis prikazuje stopnje nevarnosti za razvoj bolezni in škodljivcev vinske trte. Namenjena je predvsem populaciji, ki goji vinsko trto. Bolezni in škodljivci vinske trte v Sloveniji vsako leto prizadenejo veliko vinogradov in s tem povzročajo ogromno škodo slovenskemu vinogradništvu. S pomočjo aplikacije lahko posamezniki ali pa večja podjetja pridobijo informacije o stopnjah nevarnosti za razvoj bolezni in škodljivcev, na podlagi katerih lahko ustrezno preventivno ukrepajo in zatrejo nadaljnji razvoj. Aplikacija je zelo preprosta in za njeno uporabo ni potrebno veliko predznanja. Med samim razvojem aplikacije sem se srečal z ogromno tehničnimi orodji, ki sem jih moral združiti v celoto. Ukvarjal sem se z razvojem strežniškega dela in razvoja API-ja, razvojem mobilne aplikacije in grafičnega oblikovanja elementov, boleznimi in škodljivci vinske trte in s tem, kako se iz golih podatkov o temperaturi, tlaku, količini padavin in drugih meritev izračuna stopnjo nevarnosti za razvoj bolezni in škodljivcev. Projekt je zasnovan tako, da se lahko dodatne vremenske postaje poljubno dodajajo v projekt. Ideja projekta je, da bi podatki o boleznih bili javni in dostopni vsem ter da bi se čimveč posameznikov in podjetij odločilo za nakup vremenske postaje, saj bi bila s tem pokritost območij za razvoj bolezni vinske trte vedno večja. Ob razvoju aplikacije sem ugotovil, da izračun stopnje nevarnosti za večino bolezni in škodljivcev ni tako zapleten za strokovnjaka s področja matematike ter je za večino bolezni

objavljen tudi model, na podlagi katerega se v odvisnosti od izmerjenih vremenskih razmer izračuna stopnjo nevarnosti za razvoj bolezni. Vsi potrebni senzorji so dostopni na internetu, zato se imam namen v prihodnje lotiti izdelave lastne postaje, ki bi bila cenovno bolj ugodna. Prav tako pa bi zaradi težavnosti dostopa IPM podatkov in njihovega izvoza na centralni strežnik celotna aplikacija bolj učinkovito delovala.

Literatura

- [1] Unknown. 1.3 getting started - git basics. Dosegljivo: <https://git-scm.com/book/en/v2/Getting-Started-Git-Basics>, 2017. [Dostopano: 25.6.2017].
- [2] Unknown. About node.js. Dosegljivo: <https://nodejs.org/en/about/>, 2017. [Dostopano: 25.6.2017].
- [3] Unknown. Angularjs. Dosegljivo: <https://docs.angularjs.org/guide/introduction>, 2017. [Dostopano: 25.6.2017].
- [4] Unknown. Apidoc. Dosegljivo: <http://apidocjs.com>, 2017. [Dostopano: 23.6.2017].
- [5] Unknown. Core concepts. Dosegljivo: <http://ionicframework.com/docs/intro/concepts/>, 2017. [Dostopano: 23.6.2017].
- [6] Unknown. Ipm for grapes. Dosegljivo: http://www.davisnet.com/product_documents/weather/spec_sheets/6555_spec_WeatherLinkIP.pdf, 2017. [Dostopano: 1.8.2017].
- [7] Unknown. Scripting with windows powershell. Dosegljivo: <https://technet.microsoft.com/en-us/library/bb978526.aspx>, 2017. [Dostopano: 7.7.2017].
- [8] Unknown. Sublime text. Dosegljivo: <https://www.sublimetext.com>, 2017. [Dostopano: 23.6.2017].

- [9] Unknown. Weatherlinkip for vantage pro and vantage pro2. Dosegljivo: http://www.davisnet.com/product_documents/weather/Catalogs_Brochures/PR104.pdf, 2017. [Dostopano: 1.8.2017].
- [10] Unknown. What is mysql. Dosegljivo: <https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>, 2017. [Dostopano: 23.6.2017].
- [11] Unknown. What is node.js? Dosegljivo: https://www.tutorialspoint.com/nodejs/nodejs_introduction.htm, 2017. [Dostopano: 17.7.2017].
- [12] Stanko Vršič and Mario Lešnik. *Vinogradništvo*. ČZD Kmečki Glas, 2010.